II. CLAIM AMENDMENTS

There are no claim amendments in this response.

1. (Previously Presented) A method for routing views in a computer graphical user interface, comprising:

determining a view chain data structure comprising at least three entries, each of said entries comprising an application identifier and a view identifier, a view identified by said view identifier being within an application identified by said application identifier;

passing said view chain data structure to a view router from a first application;

detecting a first entry in said view chain data structure by said view router;

determining a first target application identifier in said first entry by said view router;

launching a first view in said first target application by calling a method associated with said first target application by said view router, said launching comprising a presentation of a first user interface form in said computer graphical user interface by said first target application;

receiving data to the first view from a user during said launching of the first view;

continuing said view router by calling a listener method associated with said view router by said first target application;

checking whether entries for views not launched remain in said view chain data structure, each said entry for a view not launched specifying a view identifier for a view not yet launched;

detecting a second entry in said view chain data structure by said view router;

determining a second target application identifier in said second entry by said view router;

launching a second view in said second target application by calling a method associated with said second target application by said view router when entries for views not launched remain in said view chain, said launching comprising a presentation of a second user interface form in said computer graphical user interface by said second target application;

receiving data to the second view from the user during said launching of the second view;

continuing said view router by calling a listener method associated with said view router by said second target application;

continuing said first application automatically when no entries for views not launched remain in said view chain data structure by calling a listener method associated with said first application by said view router.

2. (Previously Presented) The method according to claim 1, the method further comprising:

gathering data from said first view and said second view; and passing said data from said view router to said first application or to a subsequent application identified in said view chain data structure.

3. (Cancelled).

4. (Previously Presented) The method according to claim 2, wherein said gathered data is organized into a journal list comprising an entry for each view in said view chain data structure.

5. (Previously Presented) The method according to claim 2, wherein said gathered data is organized into a list of type and value pairs.

6. (Previously Presented) The method according to claim 5, wherein said data type and value pair are defined in a markup language format.

7. (Previously Presented) The method according to claim 2, wherein said view router provides a generic interface with generic methods and acts as an adapter for returning data from said first view to said first application or a subsequent application identified in said view chain data structure.

8. (Cancelled)

9. (Original) The method according to claim 1, wherein said view comprises user interface elements.

10. (Previously Presented) The method according to claim 1, wherein said view is a window opened during said launching step.

11. (Cancelled).

12. (Previously Presented) The method according to claim 1, wherein at least part of said view chain data structure is specified in a memory of an electronic device.

13. (Previously Presented) The method according to claim 12, wherein said view chain data structure is updated based on user actions.

14. (Previously Presented) The method according to claim 1, wherein said view chain data structure is determined based on user actions.

15. (Previously Presented) A system comprising:

a processor configured to:

determine a view chain data structure comprising at least three entries, each of said entries comprising an application identifier and a view identifier, a view identified by said view identifier being within an application identified by said application identifier,

pass said view chain data structure to a view router from a first application,

detect a first entry in said view chain data structure by said view router,

determine a first target application identifier in said first entry by said view router,

launch a first view in said first target application by calling a method associated with said first target application by said view router, said launching comprising a presentation of a first user interface form in said computer graphical user interface by said first target application,

receive data to the first view from a user during said launching of the first view;

continue said view router by calling a listener method associated with said view router by said first target application,

check whether entries for views not launched remain in said view chain data structure, each said entry for a view not launched specifying a view identifier for a view not yet launched,

detect a second entry in said view chain data structure by said view router,

determine a second target application identifier in said second entry by said view router,

launch a second view in said second target application by calling a method associated with said second target application by said view router when entries for views not launched remain in said view chain, said launching comprising a presentation of a second user interface form in said computer graphical user interface by said second target application,

receive data to the second view from the user during said launching of the second view;

continue said view router by calling a listener method associated with said view router by said second target application,

continue said first application automatically when no entries for views not launched remain in said view chain data structure by calling a listener method associated with said first application by said view router

; and

a display configured to display said first view and said second view.

16. (Previously Presented) The system according to claim 15, wherein said processor is further configured to gather data from said first view and said second view, and to pass said data to said first application or to a subsequent application identified in said view chain data structure.

17. (Cancelled)

18. (Previously Presented) The system according to claim 16, wherein said gathered data is organized into a journal list comprising an entry for each view in said view chain data structure.

19. (Previously Presented) The system according to claim 16, wherein said gathered data is organized into a list of type and value pairs.

20. (Original) The system according to claim 16, wherein said data type and value pair are in markup language format.

21. (Previously Presented) The system according to claim 15, wherein said view router provides a generic interface with generic methods and acts as an adapter for returning

information from said first view to said first application or a subsequent application identified in said view chain data structure.

22. (Original) The system according to claim 15, wherein said system has a graphical user interface.

23. (Original) The system according to claim 15, wherein said view comprises user interface elements.

24. (Original) The system according to claim 22, wherein said view is a window opened during view launching.

25. (Cancelled).

26. (Previously Presented) The system according to claim 15, wherein at least part of said view chain data structure is specified in the memory of an electronic device.

27. (Previously Presented) The system according to claim 26, wherein said view chain data structure is updated based on user actions.

28. (Previously Presented) The system according to claim 15, wherein said view chain data structure is determined based on user actions.

29. (Previously Presented) An apparatus comprising:

a processor configured to:

determine a view chain data structure comprising at least three entries, each of said entries comprising an application identifier and a view identifier, a view identified by said view identifier being within an application identified by said application identifier,

pass said view chain data structure to a view router from a first application,

detect a first entry in said view chain data structure by said view router,

determine a first target application identifier in said first entry by said view router,

launch a first view in said first target application by calling a method associated with said first target application by said view router, said launching comprising a presentation of a first user interface form in said computer graphical user interface by said first target application,

receive data to the first view from a user during said launching of the first view;

continue said view router by calling a listener method associated with said view router by said first target application,

check whether entries for views not launched remain in said view chain data structure, each said entry for a view not launched specifying a view identifier for a view not yet launched,

detect a second entry in said view chain data structure by said view router,

determine a second target application identifier in said second entry by said view router,

launch a second view in said second target application by calling a method associated with said second target application by said view router when entries for views not launched remain in said view chain, said launching comprising a presentation of a second user interface form in said computer graphical user interface by said second target application,

receive data to the second view from the user during said launching of the second view;

continue said view router by calling a listener method associated with said view router by said second target application,

continue said first application automatically when no entries for views not launched remain in said view chain data structure by calling a listener method associated with said first application by said view router; and

a display configured to display said first view and said second view.

30. (Previously Presented) The apparatus according to claim 29, wherein said processor is further configured to gather data from first view and said second view, and to pass said data to said first application or to a subsequent application identified in said view chain data structure.

31. (Cancelled).

32. (Previously Presented) The apparatus according to claim 30, wherein said gathered data is organized into a journal list comprising an entry for each view in said view chain data structure.

33. (Previously Presented) The apparatus according to claim 30, wherein said gathered data is organized into a list of type and value pairs.

34. (Previously Presented) The apparatus according to claim 30, wherein said data type and value pair are in markup language format.

35. (Previously Presented) The apparatus according to claim 29, wherein said view router provides a generic interface with generic methods and acts as an adapter for returning information from said first view to said first application or a subsequent application identified in said view chain data structure.

36. (Previously Presented) The apparatus according to claim 29, wherein said apparatus has a graphical user interface.

37. (Previously Presented) The apparatus according to claim 29, wherein said view comprises user interface elements.

38. (Previously Presented) The apparatus according to claim 29, wherein said view is a window opened during view launching.

39. (Cancelled).

40. (Previously Presented) The apparatus according to claim 29, wherein at least part of said view chain data structure is specified in the memory area of said electronic device.

41. (Previously Presented) The apparatus according to claim 29, wherein said view chain data structure is updated based on user actions.

42. (Previously Presented) The apparatus according to claim 29, wherein said view chain data structure is determined based on user actions.

43. (Previously Presented) A computer readable storage medium having a computer program embodied thereon, the computer program comprising code for controlling a processor to execute a method comprising:

determining a view chain data structure comprising at least three entries, each of said entries comprising an application identifier and a view identifier, a view identified by said view identifier being within an application identified by said application identifier;

passing said view chain data structure to a view router from a first application;

detecting a first entry in said view chain data structure by said view router;

determining a first target application identifier in said first entry by said view router;

launching a first view in said first target application by calling a method associated with said first target application by said view router, said launching comprising a presentation of a first user interface form in said computer graphical user interface by said first target application;

receiving data to the first view from a user during said launching of the first view;

continuing said view router by calling a listener method associated with said view router by said first target application;

checking whether entries for views not launched remain in said view chain data structure, each said entry for a view not launched specifying a view identifier for a view not yet launched;

detecting a second entry in said view chain data structure by said view router;

determining a second target application identifier in said second entry by said view router;

launching a second view in said second target application by calling a method associated with said second target application by said view router when entries for views not launched remain in said view chain, said launching comprising a presentation of a second user interface form in said computer graphical user interface by said second target application;

receiving data to the second view from the user during said launching of the second view;

continuing said view router by calling a listener method associated with said view router by said second target application; and

continuing said first application automatically when no entries for views not launched remain in said view chain data structure by calling a listener method associated with said first application by said view router.

44. (Cancelled)

45. (Previously Presented) The computer readable storage medium according to claim 43, wherein said computer readable storage medium is a removable memory card.

46. (Previously Presented) The computer readable storage medium according to claim 43, wherein said computer readable storage medium is a magnetic or optical disk.

47. (Previously Presented) The computer readable storage medium according to claim 43, comprising code for controlling a processor to execute a method further comprising:

gathering data from said at least one first view and said second view; and passing said data from said view router to said first application or to a subsequent application identified in said view chain data structure.

48. (Cancelled).

49. (Previously Presented) The computer readable storage medium according to claim 47, wherein said gathered data is organized into a journal list comprising an entry for each view in said view chain data structure.

50. (Previously Presented) The computer readable storage medium according to claim 47, wherein said gathered data is organized into a list of type and value pairs.

51. (Previously Presented) The computer readable storage medium according to claim 50, wherein said data type and value pair are in a markup language format.

52. (Previously Presented) The computer readable storage medium according to claim 47, wherein said view router provides a generic interface with generic methods and acts as an

adapter for returning data from said first view to said first application or a subsequent application identified in said view chain data structure.

53. (Previously Presented) The computer readable storage medium according to claim 43, wherein said computer program has a graphical user interface.

54. (Previously Presented) The computer readable storage medium according to claim 43, wherein said view comprises user interface elements.

55. (Previously Presented) The computer readable storage medium according to claim 53, wherein said view is a window opened during said launching step.

56. (Previously Presented) The computer readable storage medium according to claim 43, wherein at least part of said view chain data structure is specified in the memory of an electronic device.

57. (Previously Presented) The computer readable storage medium according to claim 56, wherein said view chain data structure is updated based on user actions.

58. (Previously Presented) The computer readable storage medium according to claim 43, wherein said view chain data structure is determined based on user actions.

59. - 60. (Cancelled)

61. (Previously Presented) The computer readable storage medium according to claim 43, wherein said view router is implemented as a library.

62. (Previously Presented) The computer readable storage medium according to claim 43, wherein said view router is implemented as an own application.

63. (Cancelled)